

An Alternating Direction Method for Total Variation Denoising^{*}

Zhiwei (Tony) Qin[†] Donald Goldfarb[‡] Shiqian Ma[§]

August 9, 2011

Abstract

We consider the image denoising problem using total variation regularization. This problem is computationally challenging to solve due to the non-differentiability and non-linearity of the regularization term. We propose a new alternating direction augmented Lagrangian method, involving subproblems that can be solved efficiently and exactly. We compare our method with the split Bregman method [10] and demonstrate the superiority of our method in image restoration quality on a set of standard test images.

Keywords: alternating direction method, augmented Lagrangian, split Bregman, total variation denoising

1 Introduction

In signal processing, total variation (TV) regularization is a very popular and effective approach for noise reduction and has a wide array of applications in digital imaging. The total variation is the integral of the absolute gradient of the signal. Using TV regularization to remove noise from signals was originally proposed in [21] and is based on the observation that noisy signals have high total variation. By reducing the total variation of a noisy signal while keeping the resulting signal close to the original one removes noise while preserving important details such as sharp edges. Other existing denoising techniques include median filtering and Tikhonov-like regularization, $\|u\|_{TIK} := \sum_i (\nabla_x u)_i^2 + (\nabla_y u)_i^2$. It is known that they tend to smooth away important texture details along with the noise [23, 26].

For a 2-D signal $u \in \mathbb{R}^{n \times m}$, such as an image, the total variation $\|u\|_{TV}$ [21] of u can be defined anisotropically or isotropically:

$$\|u\|_{TV} := \begin{cases} |\nabla_x u| + |\nabla_y u|, & \text{(Anisotropic);} \\ \sum_i \sqrt{(\nabla_x u)_i^2 + (\nabla_y u)_i^2}, & \text{(Isotropic).} \end{cases} \quad (1)$$

Concisely, $\|u\|_{TV}$ can be expressed as $\sum_{i=1}^{nm} \|D_i u\|_p$, where $D_i u \in \mathbb{R}^2$ denotes the discrete gradient of u at pixel i . Hence, $\|u\|_{TV}$ is isotropic when $p = 2$ and is anisotropic when $p = 1$. TV denoising (also called ROF denoising) corresponds to solving the following optimization problem,

$$\min_u \lambda \|u\|_{TV} + \frac{1}{2} \|u - b\|^2, \quad (2)$$

^{*}Affiliation: Department of Industrial Engineering and Operations Research, Columbia University, New York, NY 10027. Research supported in part by DMS 10-16571, ONR Grant N00014-08-1-1118 and DOE Grant DE-FG02-08ER25856.

[†]Email: zq2107@columbia.edu.

[‡]Email: goldfarb@columbia.edu.

[§]Email: sm2756@columbia.edu.

where $b \in \mathbb{R}^{n \times m}$ is the noisy image, and the solution u is the denoised image. $\|\cdot\|$ without a subscript denotes the l_2 -norm. Let us first focus on the case of anisotropic TV regularization. The isotropic TV model will be considered in Section 2.3. We assume that all 2-D images are in column-major vectorized form. Assuming that the equivalent one-dimensional index of (i, j) is k and $1 \leq i \leq n, 1 \leq j \leq m$, the elements of ∇u are given by

$$[\nabla u]_{ij} = \begin{pmatrix} u_{k+1} - u_k \\ u_{k+n} - u_k \end{pmatrix} = \begin{pmatrix} \nabla_x u \\ \nabla_y u \end{pmatrix}. \quad (3)$$

1.1 Related Work

Due to the non-differentiability and non-linearity of the TV term in problem (2), this problem is computationally challenging to solve despite its simple form. Hence, much effort has been devoted to devise an efficient algorithm for solving it. A number of references are provided in Section 1 of [10]. In addition, Chambolle's Algorithm [4] solves problem (2) with the isotropic TV-norm. In [26], variable-splitting and penalty concepts are applied to solving the TV-based deblurring problem

$$\min_u \lambda \sum_i \|D_i u\|_1 + \frac{1}{2} \|Ku - b\|^2. \quad (4)$$

Specifically, an auxiliary variable $w_i \in \mathbb{R}^2$ is introduced for each pixel to decouple $D_i u$ from the non-differentiable term $\sum_i \|\cdot\|_1$, yielding an approximation to problem (4)

$$\min_{w,u} \lambda \sum_i \|w_i\|_1 + \frac{\beta}{2} \sum_i \|w_i - D_i u\|^2 + \frac{1}{2} \|Ku - b\|^2. \quad (5)$$

Problem (5) is then minimized alternately with respect to w and u , with a continuation scheme driving the penalty parameter β gradually to a sufficiently large number. This method is extended in [28, 30] to solve the multi-channel (color) image deblurring problem. In [30], the TV regularization with 1-norm fidelity (TVL1) model

$$\min_u \lambda \sum_i \|D_i u\| + \|Ku - b\|_1$$

is considered. The same approach has also been applied to reconstruct signals from partial Fourier data in the compressed sensing context [31]. A downside to this quadratic penalty approach is that when β is very large, problem (5) becomes ill-conditioned and numerical stability becomes an issue.

Our algorithm is most closely related to the split Bregman method [10], which is an application of variable splitting to the Bregman iterative regularization method [16]. The Bregman iterative regularization method was first introduced in [16] as a better alternative (iterative) approach to the TV denoising problem than directly solving problem (2). The Bregman distance associated with a convex function $E(\cdot)$ between u and v is defined as

$$D_E^p(u, v) := E(u) - E(v) - p^T(u - v),$$

where $p \in \partial E(v)$ and $\partial E(v)$ denotes the subdifferential of $E(\cdot)$ at the point v . The Bregman iteration for the unconstrained minimization problem

$$\min_u E(u) + \lambda H(u),$$

where both functions $E(\cdot)$ and $H(\cdot)$ are convex, is

$$\begin{aligned} u^{(k+1)} &= \min_u D_E^p(u, u^{(k)}) + \lambda H(u) \\ &= \min_u E(u) - (u - u^{(k)})^T p^{(k)} + \lambda H(u), \end{aligned} \quad (6)$$

$$p^{(k+1)} = p^{(k)} - \nabla H(u^{(k+1)}). \quad (7)$$

If we linearize the $H(u)$ term in (6) and add an l_2 -proximity term $\frac{1}{2\mu}\|u - u^{(k)}\|^2$, we obtain the linearized Bregman iteration [33, 32]

$$u^{(k+1)} = \min_u D_E^p(u, u^{(k)}) + \lambda \nabla H(u^{(k)})^T u + \frac{1}{2\mu}\|u - u^{(k)}\|^2, \quad (8)$$

$$p^{(k+1)} = p^{(k)} - \nabla H(u^{(k)}) - \frac{1}{\mu}(u^{(k+1)} - u^{(k)}). \quad (9)$$

With the introduction of an auxiliary variable d in the spirit of [26], the TV denoising problem (2) is equivalent to the constrained problem

$$\begin{aligned} \min_{u,d} \quad & \lambda \|d\|_1 + R(u) \\ \text{s.t.} \quad & d = \Phi(u) \end{aligned} \quad (10)$$

where $R(u) = \frac{1}{2}\|u - b\|^2$, and $\Phi(u) = \begin{pmatrix} \nabla_x u \\ \nabla_y u \end{pmatrix}$. Now, converting problem (10) into an unconstrained problem (by penalizing $\|d - \Phi(u)\|^2$)

$$\min_{u,d} \lambda \|d\|_1 + R(u) + \frac{1}{2\mu}\|d - \Phi(u)\|^2$$

and applying the general Bregman iteration (6)-(7) with $E(u, d) = \lambda \|d\|_1 + R(u)$ and $H(u, d) = \|d - \Phi(u)\|^2$, we obtain after simplification the following specific Bregman iteration:

$$(u^{(k+1)}, d^{(k+1)}) = \min_{u,d} \lambda \|d\|_1 + R(u) + \frac{1}{2\mu}\|d - \Phi(u) - r^{(k)}\|^2, \quad (11)$$

$$r^{(k+1)} = r^{(k)} + (\Phi(u^{(k+1)}) - d^{(k+1)}), \quad (12)$$

with $r^{(0)} = 0$. As is well known, the Bregman iterative algorithm (11)-(12) is equivalent to applying the augmented Lagrangian method [13, 17] to solve problem (10). In [10], an approximate solution to (11) was proposed by alternatingly minimizing the right-hand-side of (11) with respect to u and d once. This yields the following split Bregman, or equivalently, alternating direction augmented Lagrangian (ADAL) algorithm (Algorithm 1.1). In this paper, we propose a different ADAL method by further splitting the vector d .

1.2 Organization of The Paper

The outline of the rest of the paper is as follows. We first describe our proposed alternating directions augmented Lagrangian method in Section 2.2. We then discuss in Section 2.4 the difference between our algorithm and the split Bregman method. In Section 3, we test both algorithms on a set of standard test images and demonstrate the superiority of our method in image restoration quality over the split Bregman method.

Algorithm 1.1 SplitBregman

```

1: Given  $u^{(0)}$ ,  $d^{(0)}$ , and  $r^{(0)}$ .
2: for  $k = 0, 1, \dots, K$  do
3:    $u^{(k+1)} \leftarrow \min_u R(u) + \frac{1}{2\mu} \|d^{(k)} - \Phi(u) - r^{(k)}\|^2$ 
4:    $d^{(k+1)} \leftarrow \min_d \lambda \|d\|_1 + \frac{1}{2\mu} \|d - \Phi(u^{(k+1)}) - r^{(k)}\|^2$ 
5:    $r^{(k+1)} \leftarrow r^{(k)} + (\Phi(u^{(k+1)}) - d^{(k+1)})$ 
6: end for
7: return  $u^{(K)}$ 

```

2 An Alternating Directions Method

Our strategy is to reformulate problem (2) into an equivalent constrained optimization problem and use the Alternating Directions Augmented Lagrangian (ADAL) method to solve the resulting problem.

2.1 ADAL

The ADAL method is also known as the alternating direction method of multipliers (ADMM) and was first proposed in the 1970s [8, 9]. It belongs to the family of the classical augmented Lagrangian (AL) method [17, 20, 13], which iteratively solves the linearly constrained problem

$$\begin{aligned} \min_x \quad & F(x) \\ \text{s.t.} \quad & Ax = b. \end{aligned} \tag{13}$$

The augmented Lagrangian of problem (13) is $\mathcal{L}(x, \gamma) = F(x) + \gamma^T(b - Ax) + \frac{1}{2\mu} \|Ax - b\|^2$, where γ is the Lagrange multiplier and μ is the penalty parameter for the quadratic infeasibility term. The AL method minimizes $\mathcal{L}(x, \gamma)$ followed by an update to γ in each iteration as stated in the following algorithm. We denote by K the user-defined maximum number of iterations or the number of iterations required to satisfy the termination criteria.

Algorithm 2.1 AL (Augmented Lagrangian method)

```

1: Choose  $\gamma^{(0)}$ .
2: for  $k = 0, 1, \dots, K$  do
3:    $x^{(k+1)} \leftarrow \arg \min_x \mathcal{L}(x, \gamma^{(k)})$ 
4:    $\gamma^{(k+1)} \leftarrow \gamma^{(k)} - \frac{1}{\mu} (Ax^{(k+1)} - b)$ 
5: end for
6: return  $x^{(K)}$ 

```

For a structured unconstrained problem

$$\min_x F(x) \equiv f(x) + g(Ax), \tag{14}$$

where both functions $f(\cdot)$ and $g(\cdot)$ are convex, we can decouple the two functions by introducing an auxiliary variable y and transform problem (14) into an equivalent linearly constrained problem

$$\begin{aligned} \min_{x,y} \quad & f(x) + g(y) \\ \text{s.t.} \quad & Ax = y. \end{aligned} \tag{15}$$

Now applying Algorithm 2.1 to the above problem, the augmented Lagrangian becomes

$$\mathcal{L}(x, y, \gamma) = f(x) + g(y) + \gamma^T(y - Ax) + \frac{1}{2\mu} \|Ax - y\|^2.$$

Exact joint minimization of $\mathcal{L}(x, y, \gamma)$ with respect to both x and y is usually difficult. In practice, an inexact version of the AL method (IAL) is often used, where $\mathcal{L}(x, y, \gamma)$ is minimized approximately as opposed to Line 3 in Algorithm 2.1. Convergence is still guaranteed in this case, as long as the subproblems are solved with an increasing accuracy [20].

ADAL (Algorithm 2.2) is an extreme case of IAL in that it finds the approximate minimizer of $\mathcal{L}(x, y, \gamma)$ by alternately optimizing with respect to x and y once. This is often desirable because joint minimization of $\mathcal{L}(x, y, \gamma)$ even approximately could be hard.

Algorithm 2.2 ADAL (ADMM)

```

1: Choose  $\gamma^{(0)}$ .
2: for  $k = 0, 1, \dots, K$  do
3:    $x^{(k+1)} \leftarrow \arg \min_x \mathcal{L}(x, y^{(k)}, \gamma^{(k)})$ 
4:    $y^{(k+1)} \leftarrow \arg \min_y \mathcal{L}(x^{(k+1)}, y, \gamma^{(k)})$ 
5:    $\gamma^{(k+1)} \leftarrow \gamma^{(k)} - \frac{1}{\mu}(Ax^{(k+1)} - y^{(k+1)})$ 
6: end for
7: return  $y^{(K)}$ 

```

Inexact versions of ADAL (IADAL), where the subproblems are solved approximately, were proposed in [6, 12]. The convergence of ADAL (and IADAL, with conditions on subproblem accuracies) has been established for the case of two-way splitting as above [6, 12]. It is also known that μ does not have to decrease to a very small value (or can simply stay constant) in order for the method to converge to the optimal solution of problem (15) [15, 3].

The versatility and simple form of ADAL have attracted much attention from a wide array of research fields. [27] applies ADAL to solve semidefinite programming problems. In signal processing/reconstruction, ADAL is applied to sparse and low-rank recovery, where nuclear norm minimization is involved [14, 34], and the l_1 -regularized problems in compressed sensing [29]. ADAL-based algorithms (SALSA and C-SALSA) [2, 1] have also been proposed to solve a number of image processing tasks, such as image inpainting and deblurring. In machine learning, ADAL and IAL-based methods have been successfully applied to the structured-sparsity estimation problems [18].

2.2 Application to TV Denoising

We consider the anisotropic TV denoising model (2). As in [10], we introduce auxiliary variables d_x and d_y for the discretized gradient components $\nabla_x u$ and $\nabla_y u$ respectively. Under the Neumann boundary condition, $\nabla_x u = Du$, where the discretization matrix $D \in \mathbb{R}^{nm \times nm}$ is an upper bidiagonal matrix with 1's on its diagonal and -1's on its super-diagonal. Similarly, $\nabla_y u = Dv$ and $v = Pu$, where P is a permutation matrix so that v is the row-major vectorized form of the 2-D image. (Recall that u is in the column-major form.) Hence, problem (2) is equivalent to the following constrained problem

$$\begin{aligned} \min_{d_x, d_y, v} \quad & \lambda(\|d_x\|_1 + \|d_y\|_1) + \frac{1}{2}\|u - b\|^2 \\ \text{s.t.} \quad & d_x = Du, \\ & d_y = Dv, \\ & v = Pu. \end{aligned} \tag{16}$$

The augmented Lagrangian of problem (16) is

$$\begin{aligned} \mathcal{L}(d, u, v, \mu) \equiv & \frac{1}{2}\|u - b\|^2 + \lambda(\|d_x\|_1 + \|d_y\|_1) + \gamma_x^T(Du - d_x) + \gamma_y^T(Dv - d_y) + \gamma_z^T(Pu - v) \\ & + \frac{1}{2\mu_1}(\|Du - d_x\|^2 + \|Dv - d_y\|^2) + \frac{1}{2\mu_2}\|Pu - v\|^2. \end{aligned} \quad (17)$$

To minimize \mathcal{L} over d , we solve

$$\min_d \lambda(\|d_x\|_1 + \|d_y\|_1) + \gamma_x^T(Du - d_x) + \gamma_y^T(Dv - d_y) + \frac{1}{2\mu_1}(\|Du - d_x\|^2 + \|Dv - d_y\|^2). \quad (18)$$

Problem (18) is separable with respect to d_x and d_y , so the unique minimizer can be computed through two soft-thresholding operations

$$\begin{aligned} d_x^* &= \mathcal{T}(Du + \mu_1\gamma_x, \lambda\mu_1), \\ d_y^* &= \mathcal{T}(Dv + \mu_1\gamma_y, \lambda\mu_1), \end{aligned}$$

where the soft-thresholding operator \mathcal{T} is defined componentwise as

$$\mathcal{T}(x, \lambda)_i := \max\{|x_i| - \lambda, 0\} \text{sign}(x_i).$$

To minimize \mathcal{L} over u , we solve

$$\min_u \frac{1}{2}\|u - b\|^2 + \gamma_x^T Du + \frac{1}{2\mu_1}\|Du - d_x\|^2 + \gamma_z^T Pu + \frac{1}{2\mu_2}\|Pu - v\|^2, \quad (19)$$

which simplifies to the linear system

$$\left(D^T D + \left(\frac{\mu_1}{\mu_2} + \mu_1\right) I\right) u = \mu_1 b + D^T(d_x - \mu_1\gamma_x) + P^T\left(\frac{\mu_1}{\mu_2}v - \mu_1\gamma_z\right) \quad (20)$$

We observe that $D^T D$ is a positive semi-definite tridiagonal matrix. Since μ_1 and μ_2 are both positive scalars, the matrix on the left-hand-side of the above system is positive definite tridiagonal. Linear systems of this special structure can be solved efficiently by the Thomas algorithm in $8nm$ flops [11]. We denote the solution to the above tridiagonal system by $u(d_x, v, \gamma_x, \gamma_z)$.

Similarly, the sub-problem with respect to v simplifies to the tridiagonal system

$$\left(D^T D + \frac{\mu_1}{\mu_2} I\right) v = D^T(d_y - \mu\gamma_y) + \mu\gamma_z + \frac{\mu_1}{\mu_2} Pu. \quad (21)$$

Its solution is denoted by $v(d_y, v, \gamma_y, \gamma_z)$.

With all the ingredients of the algorithm explained, we formally state the ADAL algorithm below.

2.3 The Isotropic Case

The isotropic TV denoising model differs from the anisotropic model in the definition of the TV norm. In this case, we define $\|u\|_{TV}^{ISO} := \sum_i \sqrt{(\nabla_x u)_i^2 + (\nabla_y u)_i^2}$, and the optimization problem to solve is

$$\min_u \lambda \|u\|_{TV}^{ISO} + \frac{1}{2}\|u - b\|^2. \quad (22)$$

Algorithm 2.3 ADAL (TV denoising)

```

1: Given  $u^{(0)}, v^{(0)}, d^{(0)}, \lambda, \gamma_x^{(0)}, \gamma_y^{(0)}, \gamma_z^{(0)}$ .
2: for  $k = 0, 1, \dots, K$  do
3:    $d_x^{(k+1)} \leftarrow \mathcal{T}(Du^{(k)} + \mu_1 \gamma_x^{(k)}, \lambda \mu_1)$ 
4:    $d_y^{(k+1)} \leftarrow \mathcal{T}(Dv^{(k)} + \mu_1 \gamma_y^{(k)}, \lambda \mu_1)$ 
5:    $v^{(k+1)} \leftarrow v(d_y^{(k+1)}, u^{(k)}, \gamma_y^{(k)}, \gamma_z^{(k)})$ , the solution of (21)
6:    $u^{(k+1)} \leftarrow u(d_x^{(k+1)}, v^{(k+1)}, \gamma_x^{(k)}, \gamma_z^{(k)})$ , the solution of (20)
7:    $\gamma_x^{(k+1)} \leftarrow \gamma_x^{(k)} + \frac{1}{\mu_1}(Du^{(k+1)} - d_x^{(k+1)})$ 
8:    $\gamma_y^{(k+1)} \leftarrow \gamma_y^{(k)} + \frac{1}{\mu_1}(Dv^{(k+1)} - d_y^{(k+1)})$ 
9:    $\gamma_z^{(k+1)} \leftarrow \gamma_z^{(k)} + \frac{1}{\mu_2}(Pu^{(k+1)} - v^{(k+1)})$ 
10: end for
11: return  $u^{(K)}$ 

```

We observe that by the definition above, $\|u\|_{TV}^{ISO} = \sum_i \|([\nabla_x u]_i, [\nabla_y u]_i)\|$, which is the group lasso regularization on $(\nabla_x u, \nabla_y u)$, with each group consisting of $([\nabla_x u]_i, [\nabla_y u]_i)$. We introduce the same auxiliary variables and linear constraints among them as in the previous section. As a result, the two subproblems with respect to u and v are the same as (20) and (21) respectively. The subproblem with respect to d now becomes

$$\min_d \lambda \sum_i \|([d_x]_i, [d_y]_i)\| + \gamma_x^T (Du - d_x) + \gamma_y^T (Dv - d_y) + \frac{1}{2\mu_1} (\|Du - d_x\|^2 + \|Dv - d_y\|^2), \quad (23)$$

which is a proximal problem associated with the group $l_{1,2}$ -norm $\|d\|_{1,2}$ with $d_x \equiv \nabla_x u, d_y \equiv \nabla_y u$, and the groups being defined above. The solution to this subproblem is thus given by a block soft-thresholding operation [25, 19, 5], $\mathcal{S}\left(\begin{pmatrix} D & 0 \\ 0 & D \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \mu_1 \begin{pmatrix} \gamma_x \\ \gamma_y \end{pmatrix}, \lambda \mu_1\right)$, where the block soft-thresholding operator is defined blockwise as

$$\mathcal{S}(x, \lambda)_i := \max\{\|x_i\| - \lambda, 0\} \frac{x_i}{\|x_i\|},$$

and x_i is the i -th block of x , i.e. $([Du]_i, [Dv]_i)$ in our case.

2.4 Comparison with The Split Bregman Method

The Bregman iterative regularization method has been shown to be equivalent to the classical augmented Lagrangian method [33]. Moreover, the alternating minimization approach for minimizing the subproblem (11) makes the split Bregman method (Algorithm 1.1) equivalent to Algorithm 2.2 [24, 7, 22] applied to the constrained problem

$$\begin{aligned} \min_{d,u} \quad & \lambda(\|d_x\|_1 + \|d_y\|_1) + \frac{1}{2}\|u - b\|^2 \\ \text{s.t.} \quad & d_x = \nabla_x u, \\ & d_y = \nabla_y u. \end{aligned}$$

It is clear now that the main difference between Algorithm 2.3 (ADAL) and the split Bregman method is the additional constraint $v = Pu$ in problem (16). The split Bregman subproblem with respect to u (line 3 in Algorithm 1.1) can be simplified to the linear system

$$(\mu I + (\nabla_x^T \nabla_x + \nabla_y^T \nabla_y)) u^{(k+1)} = \mu b + \nabla_x^T (d_x^{(k)} - r_x^{(k)}) + \nabla_y^T (d_y^{(k)} - r_y^{(k)}), \quad (24)$$



Figure 1: The set of standard test images.

whose left-hand-side matrix includes a Laplacian matrix and is strictly diagonally dominant. Solving this linear system exactly in each iteration is expensive. Hence, one iteration of the Gauss-Seidel method is applied in [10] to solve (24) approximately.

In contrast, the subproblems with respect to v and u ((21) and (20)) in ADAL (Algorithm 2.3) have much better structures and thus can be solved exactly in an efficient manner as we saw in Section 2.2. We surmised that this is a better approach for the TV denoising problem; the results in the next section confirmed this.

3 Experiments

Our ADAL algorithm (Algorithm 2.3) was written in Matlab, whereas SplitBregman is in C with a Matlab interface.¹ We ran both algorithms on a laptop with an Intel Core 2 Duo processor and 4G memory.

3.1 Test images

We compared our ADAL algorithm with the split Bregman method on a set of six standard test images: **lena**, **cameraman**, **house**, **blonde**, **mandril**, and **peppers**. They present a range of challenges to image denoising algorithms, such as the reproduction of fine detail and textures, sharp transitions and edges, and uniform regions. Each image is 512×512 in grey-scale pixels and is denoted by u_0 in vectorized form. The original images are presented in Figure 1.

¹Code downloaded from <http://www.stanford.edu/tagoldst/code.html>.

3.2 Set-up

We constructed the noisy image by adding Gaussian noise to the original image, i.e. $b = u_0 + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma)$ and b is the vectorized noisy image. The quality of the denoised image in the k -th iteration, $u^{(k)}$ is measured by the normalized error with respect to the original noiseless image, i.e. $\eta^{(k)} = \frac{\|u^{(k)} - u_0\|}{\|u_0\|}$. For the final images returned by the denoising algorithms, we used the peak-signal-to-noise ratio (PSNR) to compare the reconstruction quality. The PSNR of an image u with respect to the noiseless image u_0 , in the case where the maximum pixel magnitude is 255, is defined as

$$PSNR = 20 \log_{10} \left(\frac{255\sqrt{nm}}{\|u - u_0\|} \right).$$

Note that the PSNR is monotone decreasing with the normalized error, i.e. a higher PSNR indicates better reconstruction quality. We tested three cases with the anisotropic model: low noise ($\sigma = 30$), medium noise ($\sigma = 50$), and high noise ($\sigma = 75$) with a range of values for λ . In order to select the appropriate λ 's for the different levels of noise, we considered both the PSNR and visual impression. Although the PSNR is a good indicator of reconstruction quality, it is not always true that a denoised image with a higher PSNR with respect to the original image is more visually appealing than another with a lower PSNR. Hence, for each of the three σ 's, we examine in the next section three values of λ : (1) one that gives the best visual quality, (2) one considerably smaller than (1), and (3) one considerably larger than (1).

3.3 Strategy for updating μ

We started with $\mu_1 = 0.2$ and $\mu_2 = 1.5$. We decreased μ_1 in (17) by a factor of $\beta = 0.5$ and μ_2 by a factor of $\beta = 0.1$ if $\eta^{(k)}$ increased for $K_\mu = 3$ consecutive iterations. The lower limit of μ_1 and μ_2 was set to 10^{-4} .

3.4 Results

In all test cases, we ran ADAL and SplitBregman for 25 iterations, at which point both algorithms have converged. We report the best PSNR's obtained for the six images at each of the noise levels in Table 1. The first striking observation is that ADAL outperformed SplitBregman in all the test cases in the table. In fact, even if we consider the other values of λ , ADAL *always* produced a higher PSNR than SplitBregman. (See Figures 2, 3, and 4.) We also present the convergence plots of the normalized errors for all test images. To avoid repetition, we show the plots only in the case of medium noise. Basically, the graphs show that ADAL has a very competitive convergence speed compared to SplitBregman. Considering that the per-iteration computational cost of ADAL is only slightly more expensive than the per-iteration cost of SplitBregman, the total denoising times are expected to be similar.

3.4.1 λ for best visual quality

We observed in the experiments that the best visual quality for a given image and method usually occurred at a λ slightly larger than the one giving the highest PSNR, which tended to appear noisier than one expects. Specifically, we found that setting $\lambda = 30$ for low noise, $\lambda = 60$ for medium noise, and $\lambda = 80$ (for **mandril**, $\lambda = 70$) for high noise gave the best visual results. The gaps in the normalized error and PSNR are quite small (Figure 5), but there are subtle differences in the reconstruction quality as we illustrate in Figure 6. We have magnified the restored images of **lena** around the face. Besides a higher PSNR, the image produced by ADAL clearly appears

σ	Algs	lena	cameraman	mandril	blonde	house	peppers
30	ADAL	29.299	29.836	25.481	27.919	32.094	29.515
	SplitBregman	29.125	29.615	25.115	27.893	31.815	29.273
50	ADAL	27.042	27.138	23.493	25.735	29.984	27.495
	SplitBregman	26.910	27.062	23.327	25.630	29.814	27.342
75	ADAL	25.845	25.750	22.157	24.766	28.216	26.179
	SplitBregman	25.731	25.724	22.028	24.699	28.215	26.055

Table 1: The highest PSNR's produced by ADAL and SplitBregman on each of the test images with different amount of noise.

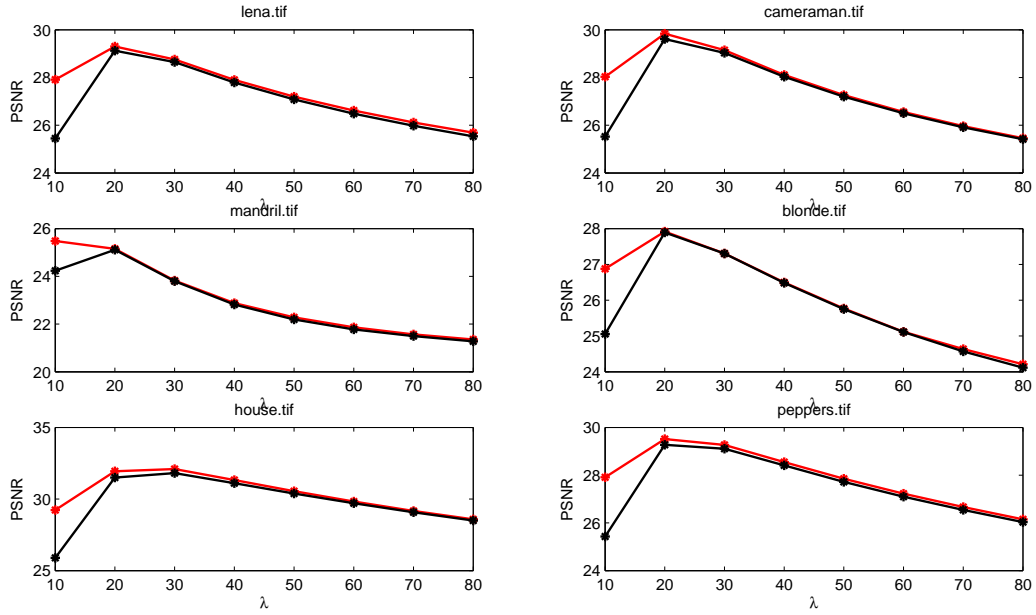


Figure 2: Plots of PSNR against λ for $\sigma = 30$. Red: ADAL. Black: SplitBregman.

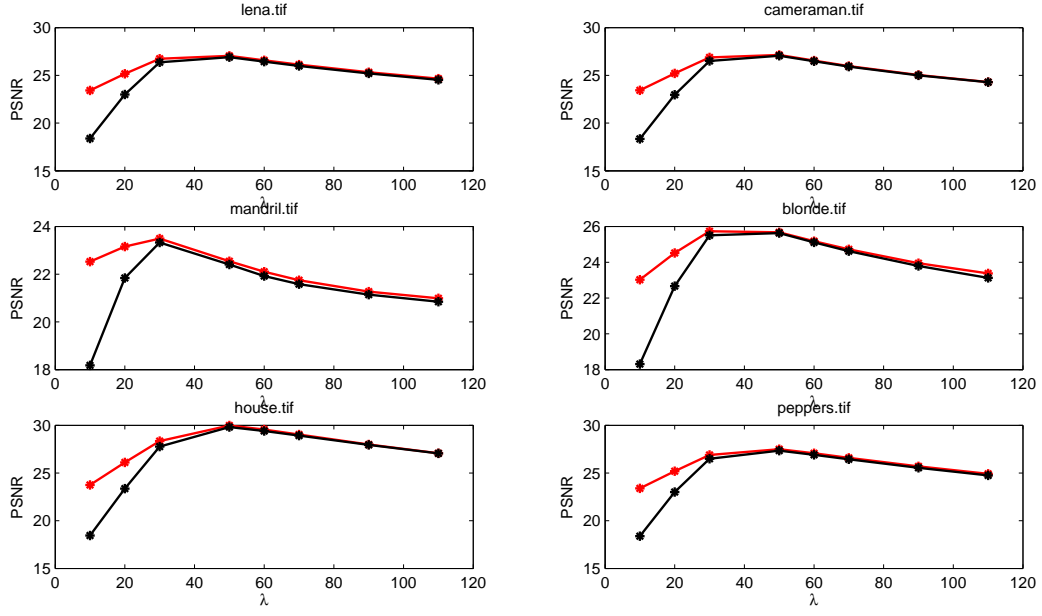


Figure 3: Plots of PSNR against λ for $\sigma = 50$. Red: ADAL. Black: SplitBregman.

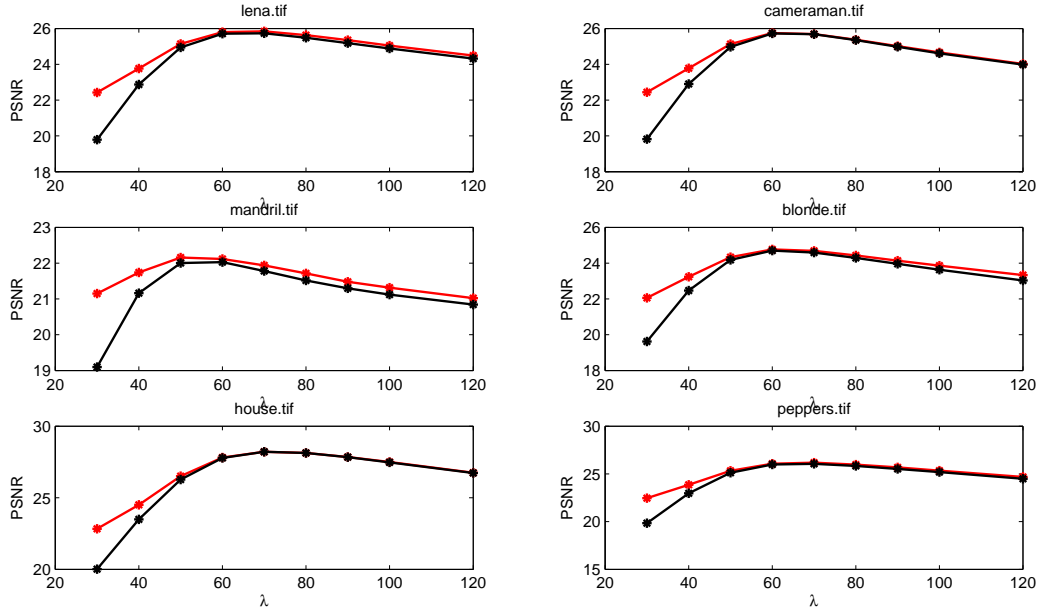


Figure 4: Plots of PSNR against λ for $\sigma = 75$. Red: ADAL. Black: SplitBregman.

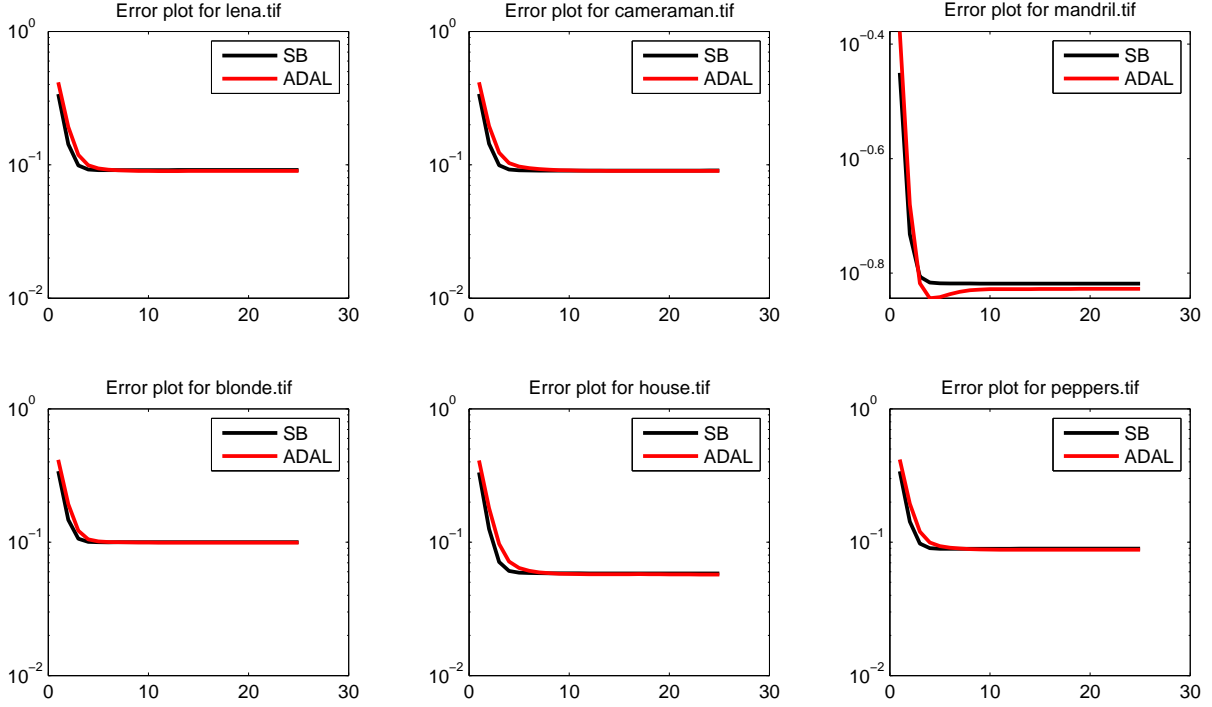


Figure 5: Convergence plots of normalized errors for $\sigma = 50$ and $\lambda = 60$.

more appealing. In the case of **mandril**, there are also tangible improvements in contrast and texture details (e.g. the nose) in the results from ADAL. For images with more or less piecewise constant intensities (e.g. **house** and **cameraman**), the two algorithms rendered very similar and good reconstruction results.

3.4.2 Small λ

For small λ 's, we set $\lambda = 10$ for low/medium noise and $\lambda = 30$ for high noise. Figure 7 shows that ADAL leads SplitBregman by a significant margin in terms of normalized error (and hence, PSNR as well). We also present the final denoising results after 25 iterations on three test images in Figure 8. Since small λ 's correspond to low regularization power, as expected the resulting images still appear fairly noisy. However, the images produced by ADAL clearly have better contrasts than those produced by SplitBregman.

3.4.3 Large λ

In the case of large λ 's, we set $\lambda = 80$ for low noise, $\lambda = 110$ for medium noise, and $\lambda = 120$ for high noise. From the denoised images produced by both algorithms, we see that there is an over-smoothing effect due to the large TV regularizations. Although the normalized errors achieved by the two algorithms are quite close (Figure 9), the difference in the image restoration quality is apparent in Figure 10. ADAL was able to render a smooth surface in areas with a gradual color transition (e.g. the pepper skin and the women's faces) while maintaining sharp edges at the boundaries of different objects. In contrast, the images produced by SplitBregman appear patchy

noisy image, $\sigma=30$ $\lambda=30$ PSNR= 18.590



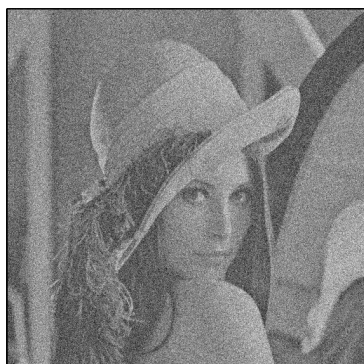
ADAL, PSNR= 32.094



SplitBregman, PSNR= 31.815



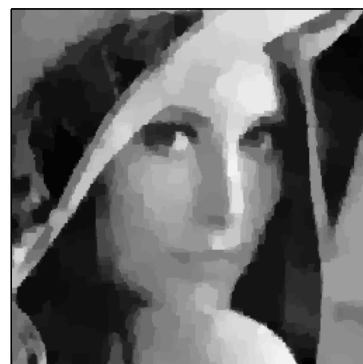
noisy image, $\sigma=50$ $\lambda=60$ PSNR= 14.165



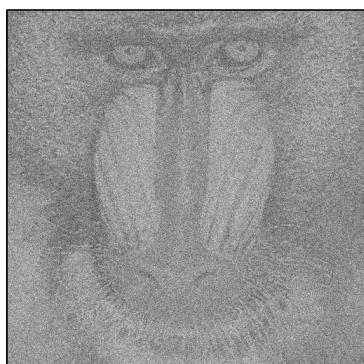
ADAL, PSNR= 26.579



SplitBregman, PSNR= 26.435



noisy image, $\sigma=75$ $\lambda=70$ PSNR= 10.612



ADAL, PSNR= 21.937



SplitBregman, PSNR= 21.780



Figure 6: Comparison of reconstruction quality for best λ 's. The denoised images of **lena** have been zoomed in around the face.

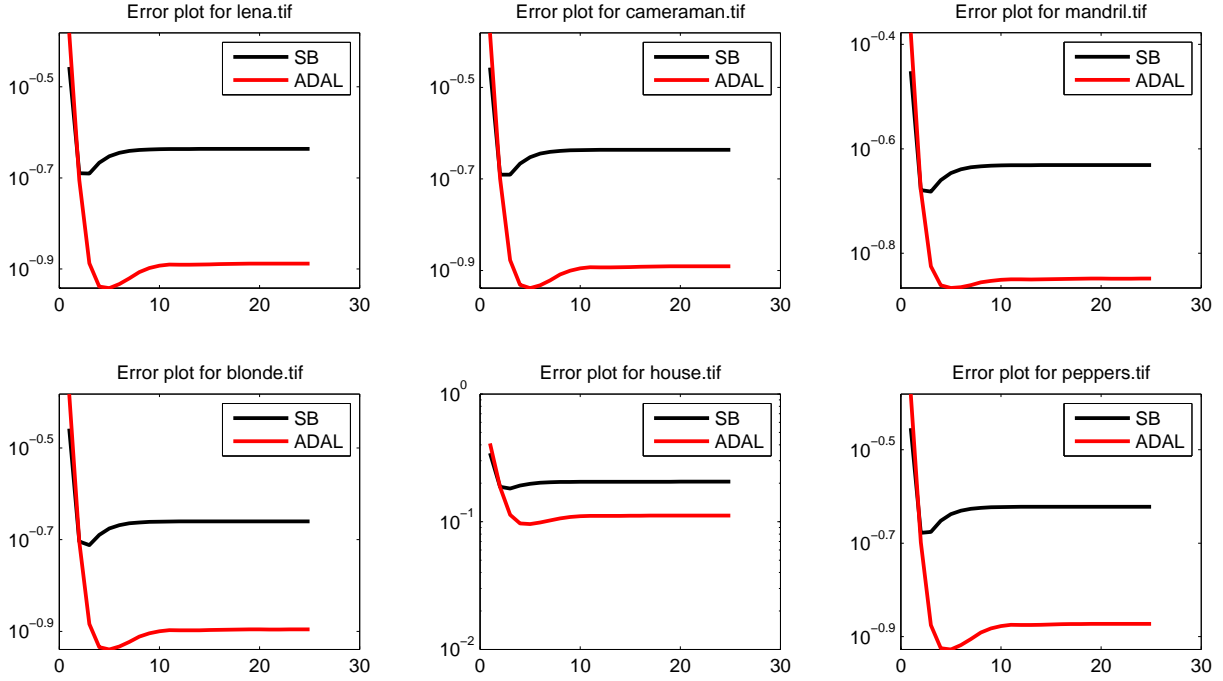


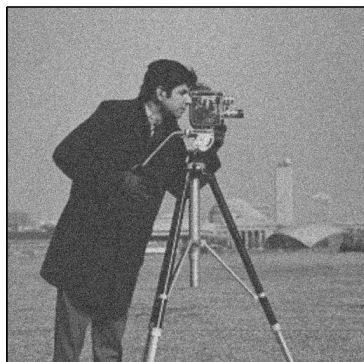
Figure 7: Convergence plots of normalized errors for $\sigma = 50$ and $\lambda = 10$.

and leave an impression of flatness. This is because there are unnecessary sharp edges in areas of color gradients, e.g. shades.

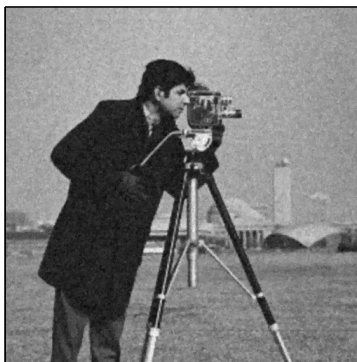
References

- [1] M. Afonso, J. Bioucas-Dias, and M. Figueiredo. An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems. *Image Processing, IEEE Transactions on*, (99):1–1, 2009.
- [2] M. Afonso, J. Bioucas-Dias, and M. Figueiredo. Fast image recovery using variable splitting and constrained optimization. *Image Processing, IEEE Transactions on*, 19(9):2345–2356, 2010.
- [3] D. Bertsekas. *Nonlinear programming*. Athena Scientific Belmont, MA, 1999.
- [4] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1):89–97, 2004.
- [5] P. Combettes and J. Pesquet. Proximal splitting methods in signal processing. *Arxiv preprint arXiv:0912.3522*, 2009.
- [6] J. Eckstein and D. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1):293–318, 1992.

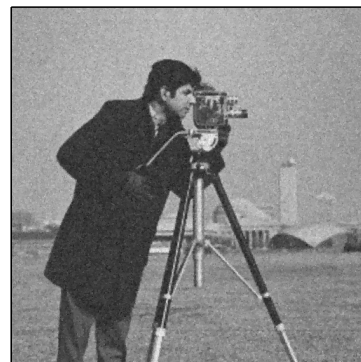
noisy image, $\sigma=30$ $\lambda=10$ PSNR= 18.590



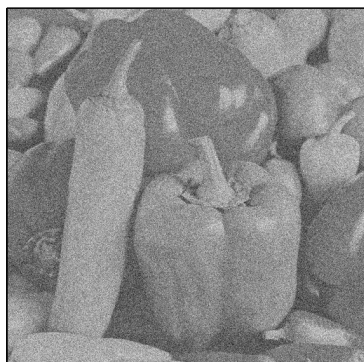
ADAL, PSNR= 28.030



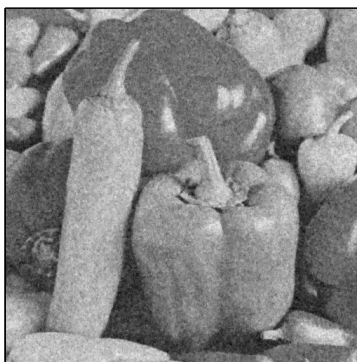
SplitBregman, PSNR= 25.521



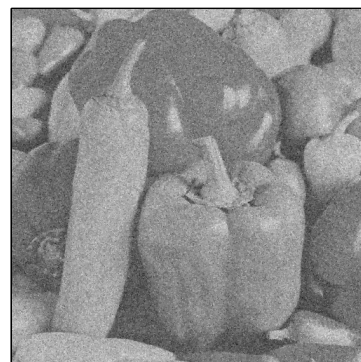
noisy image, $\sigma=50$ $\lambda=10$ PSNR= 14.163



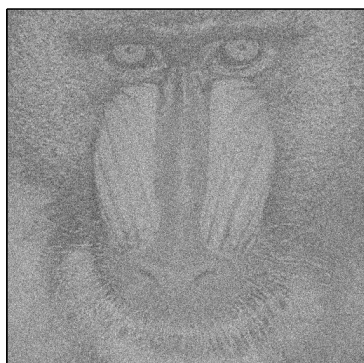
ADAL, PSNR= 23.397



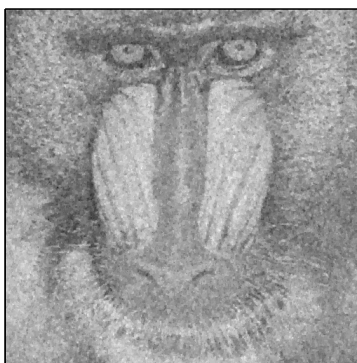
SplitBregman, PSNR= 18.382



noisy image, $\sigma=75$ $\lambda=30$ PSNR= 10.612



ADAL, PSNR= 21.148



SplitBregman, PSNR= 19.097

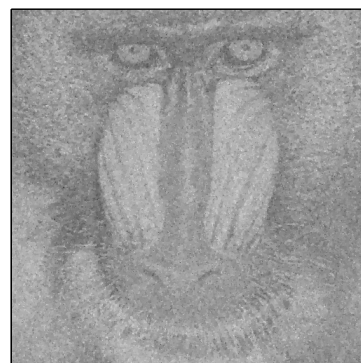


Figure 8: Comparison of reconstruction quality for small λ 's.

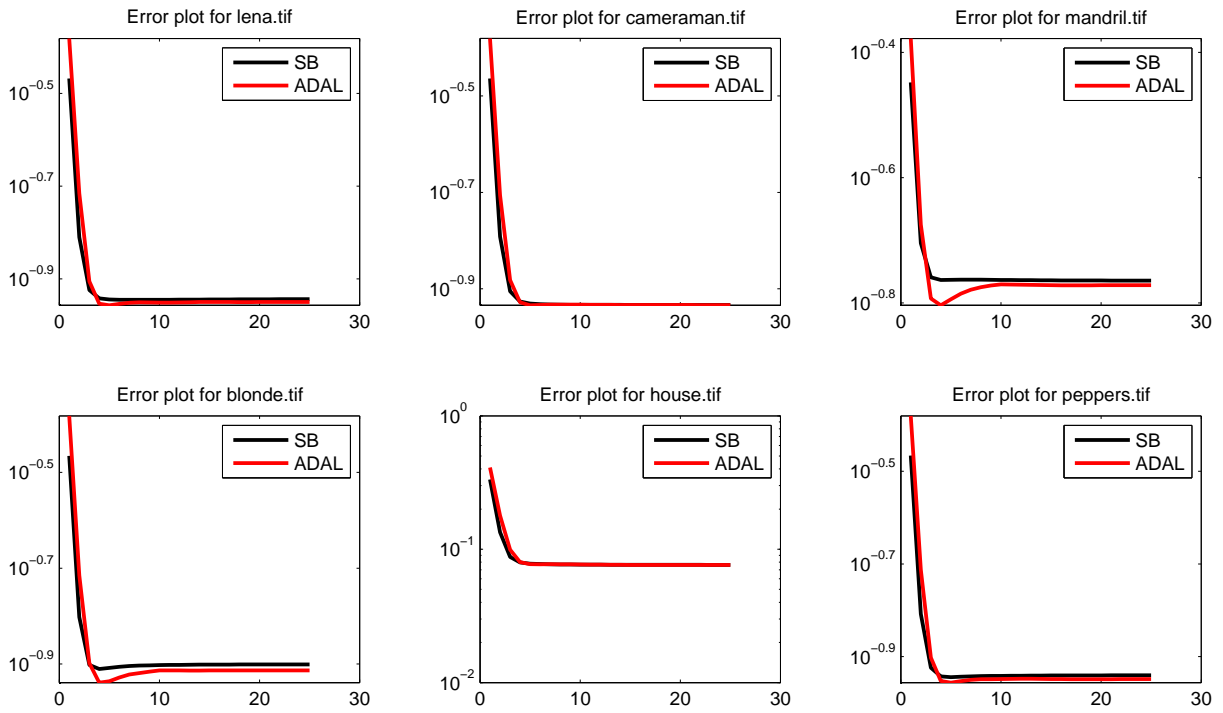


Figure 9: Convergence plots of normalized errors for $\sigma = 50$ and $\lambda = 110$.

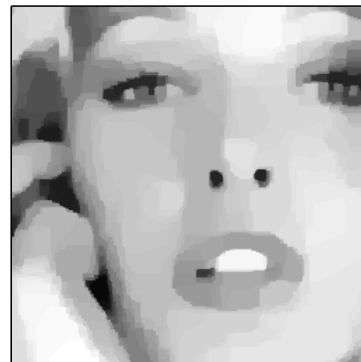
noisy image, $\sigma=30$ $\lambda=80$ PSNR= 18.566



ADAL, PSNR= 24.211



SplitBregman, PSNR= 24.116



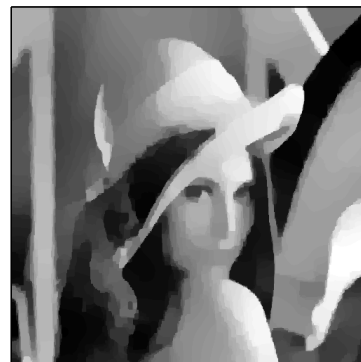
noisy image, $\sigma=50$ $\lambda=110$ PSNR= 14.165



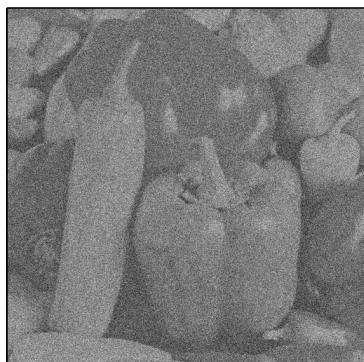
ADAL, PSNR= 24.657



SplitBregman, PSNR= 24.533



noisy image, $\sigma=75$ $\lambda=120$ PSNR= 10.614



ADAL, PSNR= 24.679



SplitBregman, PSNR= 24.493



Figure 10: Comparison of reconstruction quality for large λ 's. The denoised images of **blonde** have been zoomed in.

- [7] E. Esser. Applications of lagrangian-based alternating direction methods and connections to split bregman. *CAM report*, 9:31, 2009.
- [8] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40, 1976.
- [9] R. Glowinski and A. Marroco. Sur l’approximation, par elements finis d’ordre un, et la resolution, par penalisation-dualite d’une classe de problemes de dirichlet non lineares. *Rev. Francaise d’Automat. Inf. Recherche Operationelle*, (9):41–76, 1975.
- [10] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Sciences*, 2:323, 2009.
- [11] G. Golub and C. Van Loan. *Matrix computations*. Johns Hopkins Univ Pr, 1996.
- [12] B. He, L. Liao, D. Han, and H. Yang. A new inexact alternating directions method for monotone variational inequalities. *Mathematical Programming*, 92(1):103–118, 2002.
- [13] M. Hestenes. Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320, 1969.
- [14] Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *Arxiv preprint arXiv:1009.5055*, 2010.
- [15] J. Nocedal and S. Wright. *Numerical optimization*. Springer verlag, 1999.
- [16] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling and Simulation*, 4(2):460–489, 2006.
- [17] M. Powell. A method for nonlinear constraints in minimization problems. In R. Fletcher, editor, *Optimization*. Academic Press, New York, New York, 1972.
- [18] Z. Qin and D. Goldfarb. Structured Sparsity via Alternating Directions Methods. *ArXiv e-prints*, May 2011.
- [19] Z. Qin, K. Scheinberg, and D. Goldfarb. Efficient Block-coordinate Descent Algorithms for the Group Lasso. 2010.
- [20] R. Rockafellar. The multiplier method of hestenes and powell applied to convex programming. *Journal of Optimization Theory and Applications*, 12(6):555–562, 1973.
- [21] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.
- [22] S. Setzer. Split bregman algorithm, douglas-rachford splitting and frame shrinkage. *Scale space and variational methods in computer vision*, pages 464–476, 2009.
- [23] D. Strong and T. Chan. Edge-preserving and scale-dependent properties of total variation regularization. *Inverse problems*, 19:S165, 2003.
- [24] X. Tai and C. Wu. Augmented lagrangian method, dual methods and split bregman iteration for rof model. *Scale Space and Variational Methods in Computer Vision*, pages 502–513, 2009.

- [25] E. van den Berg, M. Schmidt, M. Friedlander, and K. Murphy. Group sparsity via linear-time projection. Technical report, Technical Report TR-2008-09, Department of Computer Science, University of British Columbia, 2008.
- [26] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272, 2008.
- [27] Z. Wen, D. Goldfarb, and W. Yin. Alternating direction augmented lagrangian methods for semidefinite programming. *Mathematical Programming Computation*, pages 1–28, 2010.
- [28] J. Yang, W. Yin, Y. Zhang, and Y. Wang. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM Journal on Imaging Sciences*, 2(2):569–592, 2009.
- [29] J. Yang and Y. Zhang. Alternating direction algorithms for l1-problems in compressive sensing. *Arxiv preprint arXiv:0912.1185*, 2009.
- [30] J. Yang, Y. Zhang, and W. Yin. An efficient tvl1 algorithm for deblurring multichannel images corrupted by impulsive noise. *SIAM J. Sci. Comput*, 31(4):2842–2865, 2009.
- [31] J. Yang, Y. Zhang, and W. Yin. A fast alternating direction method for tvl1-l2 signal reconstruction from partial fourier data. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):288–297, 2010.
- [32] W. Yin. Analysis and generalizations of the linearized bregman method. *Submitted to SIAM Journal Imaging Sciences*, 2009.
- [33] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman iterative algorithms for l1-minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1(1):143–168, 2008.
- [34] X. Yuan and J. Yang. Sparse and low-rank matrix decomposition via alternating direction methods. *preprint*, 2009.